**Codes for Analyses:**

**The instructions herein can be considered as README.**

**Single Variant association:**

We used ‘rvtest’ for single variant association. Given below is an example code snippet that we used when the response phenotype of interest was fluid intelligence score. “All\_Cog\_Pheno.ped” is a ped file with the first five columns as a standard ped file (columns- FID, IID,PID, MID, SEX). All other covariates were placed sixth column onwards. Final\_exome\_auto\_chr\_renamed.vcf.gz is the vcf file that is obtained after quality control. The tabix-indexed file should also be kept in the same folder.

*rvtest -inVcf Final\_exome\_auto\_chr\_renamed.vcf.gz --pheno All\_Cog\_Pheno\_new.ped --pheno-name FIS --covar All\_Cog\_Pheno\_new.ped --covar-name Gen\_Sex,Age,CollegeEdu,ALevelEdu,APOE\_Risk,APOE\_Beneficial,PC1,PC2,PC3,PC4,PC5,PC6,PC7,PC8,PC9,PC10 --inverseNormal --useResidualAsPhenotype --impute hwe --single Wald --out Out\_FIS\_Baseline –numThread 70*

Here ‘CollegeEdu’ and ‘ALevelEdu’ are dummies coding for the education categories, and ‘*APOE*\_Risk’ and ‘*APOE*\_Beneficial’ are dummies coding for APOE carrier status respectively.

The output is stored in ‘*Out\_FIS\_Baseline.SingleWald.assoc’* and ‘*Out\_FIS\_v30.SkatO.assoc’*

For models controlling for additional lipid/glycemic risks, say glucose we use the following code:

*rvtest -inVcf Final\_exome\_auto\_chr\_renamed.vcf.gz --pheno All\_Cog\_Pheno\_new.ped --pheno-name FIS --covar All\_Cog\_Pheno\_new.ped --covar-name Gen\_Sex,Age,CollegeEdu,ALevelEdu,APOE\_Risk,APOE\_Beneficial,PC1,PC2,PC3,PC4,PC5,PC6,PC7,PC8,PC9,PC10,Glucose --inverseNormal --useResidualAsPhenotype --impute hwe --single Wald --out Out\_FIS\_Glucose –numThread 70*

For models, which λGC >1.1, we used the following code to get the GC-corrected p-values and obtained the significant hits.

*data=read.table("Out\_FIS\_Glucose.SingleWald.assoc",header=TRUE)*

*Pval=data$Pvalue*

*lambda=qchisq(median(na.omit(data$Pvalue)),1,lower.tail=FALSE)/qchisq(0.5,1)*

*print(paste("FIS-Glucose lambda=",lambda))*

*data$Pvalue\_adj=pchisq((qchisq(data$Pvalue,1,lower.tail=FALSE)/lambda),df=1,lower.tail=FALSE)*

*write.table(data,"Out\_FIS\_Glucose\_GC.SingleWald.assoc",row.names=FALSE,quote=FALSE,sep="\t")*

*hits=data[which(data$Pvalue<alpha),]*

*hits\_GC=data[which(data$Pvalue\_adj<alpha),]*

*write.table(hits,"FIS\_Glucose.SingleWald.hits.txt",row.names=FALSE,quote=FALSE)*

*write.table(hits\_GC,"FIS\_Glucose\_GC.SingleWald.hits.txt",row.names=FALSE,quote=FALSE)*

To generate Manhattan and QQplots, we used the following sample R code:

*#FIS*

*FIS\_baseline=read.table("Out\_FIS\_GC.SingleWald.assoc",header=TRUE)*

*colnames(FIS\_baseline)[10]="Pvalue\_adj\_Baseline"*

*FIS\_HDL=read.table("Out\_FIS\_HDL.SingleWald.assoc",header=TRUE)*

*colnames(FIS\_HDL)[9]="Pvalue\_HDL"*

*FIS\_GLUC=read.table("Out\_FIS\_Glucose.SingleWald.assoc",header=TRUE)*

*colnames(FIS\_GLUC)[9]="Pvalue\_GLUC"*

*#install.packages("dplyr")*

*library(dplyr)*

*FIS=data.frame()*

*FIS=cbind(select(FIS\_baseline,c("CHROM","POS","Test","Pvalue\_adj\_Baseline")),select(FIS\_HDL,c("Pvalue\_HDL")),select(FIS\_GLUC,c("Pvalue\_GLUC")))*

*head(FIS)*

*#Annotating significant SNPs*

*FIS$SNP=array()*

*FIS$SNP[which(FIS$Test=="5:141185287")]=paste("rs115865641")*

*FIS$SNP[which(FIS$Test=="7:95406923")]=paste("rs17876162")*

*FIS$SNP[which(FIS$Test=="10:92240093")]=paste("rs3824734")*

*head(FIS)*

*FIS\_v1=FIS[,c(7,1,2,4,5,6)]*

*head(FIS\_v1)*

*library(CMplot)*

*SNPs <- list(*

*FIS\_v1$SNP[FIS$Test=="5:141185287"],*

*FIS\_v1$SNP[which((FIS$Test=="7:95406923") |(FIS$Test=="10:92240093"))],*

*FIS\_v1$SNP[which((FIS$Test=="7:95406923") |(FIS$Test=="10:92240093"))])*

*SNPs*

*colnames(FIS\_v1)[c(4,5,6)]=c("FIS Baseline" ,"FIS-HDL controlled","FIS-Glucose controlled")*

*#Multi-track Manhattan plots*

*CMplot(FIS\_v1, plot.type=c("m","q"),multracks=TRUE,points.alpha=255,threshold=2.5e-7,threshold.lty=1,*

*threshold.lwd=1, threshold.col=c("black"), amplify=TRUE,bin.size=1e6,signal.col=NULL,*

*signal.cex=1, file="jpg",memo="",dpi=300,file.output=TRUE,verbose=TRUE,*

*highlight=SNPs,highlight.col=NULL, highlight.text=SNPs,highlight.text.cex = 2,*

*highlight.text.font = 2,*

*col=matrix(c("red","#FFCCCC",NA,"deep sky blue","#ADD8E6",NA,"#4B0082","#E5CCFF",NA),*

*3,3,byrow=T))*

*#Multi-QQ plots*

*CMplot(FIS\_v1, plot.type=c("m","q"),multracks=TRUE,points.alpha=255,threshold=2.5e-7,threshold.lty=1,*

*threshold.lwd=1, threshold.col=c("black"), amplify=TRUE,bin.size=1e6,signal.col=NULL,*

*signal.cex=1, file="jpg",memo="",dpi=300,file.output=TRUE,verbose=TRUE,*

*highlight=SNPs,highlight.col=NULL, highlight.text=SNPs,*

*col=c("red","deep sky blue","#4B0082"))*

**Gene-based association:**

For our pathway-specific analyses we selected 20 genes and tested them. The example given below shows the context specific gene-based analysis for fluid intelligence score baseline model)

*rvtest -inVcf Final\_exome\_auto\_chr\_renamed.vcf.gz --pheno All\_Cog\_Pheno\_new.ped --pheno-name FIS --covar All\_Cog\_Pheno\_new.ped --covar-name Gen\_Sex,Age,CollegeEdu,ALevelEdu,APOE\_Risk,APOE\_Beneficial,PC1,PC2,PC3,PC4,PC5,PC6,PC7,PC8,PC9,PC10 --inverseNormal --useResidualAsPhenotype --impute hwe --geneFile ../../refFlat\_hg38.txt.gz --gene TREM2,APP,LRP1,SORL1,PSEN1,LDLR,LRP8,SCARB1,LCAT,ABCA1,ANGPTL3,NR1H4,APOC1,APOBR,LRP10,MAPT,LGALS1,PDCD4,HP,SNCA --kernel skat,skato --out Out\_FIS\_v30 --numThread 70*

The output is stored in ‘*Out\_FIS\_v30.Skat.assoc’* and ‘*Out\_FIS\_v30.SkatO.assoc’*

For our exome wide gene-based analysis, we used the following code:

*rvtest -inVcf Final\_exome\_auto\_chr\_renamed.vcf.gz --pheno All\_Cog\_Pheno\_new.ped --pheno-name FIS --covar All\_Cog\_Pheno\_new.ped --covar-name Gen\_Sex,Age,CollegeEdu,ALevelEdu,APOE\_Risk,APOE\_Beneficial,PC1,PC2,PC3,PC4,PC5,PC6,PC7,PC8,PC9,PC10 --inverseNormal --useResidualAsPhenotype --impute hwe --geneFile refFlat\_hg38.txt.gz --kernel skat,skato --out Out\_FIS\_v30\_ALL --numThread 70*

The output is stored in ‘*Out\_FIS\_v30\_ALL.Skat.assoc’* and ‘*Out\_FIS\_v30\_ALL.SkatO.assoc’*

**Interaction analysis:**

For interaction analysis with *APOE* variants, we need a set file with the variant sets among which we need to test for interaction.

For fluid intelligence with the baseline model, we have our set file as

*19 44908684 44908684 APOE*

*19 44908822 44908822 APOE*

*5 141185287 141185287 SET2*

where the two *APOE* variants form a set and the single variant hit from fluid intelligence baseline model form the second set.

At first, we extract the variants among which we want to find interaction.

*plink --fam v30\_INT\_Resid\_FIS.fam --bim Final\_exome\_auto\_chr\_renamed.bim --bed Final\_exome\_auto\_chr\_renamed.bed --extract range Set\_file.txt --make-bed --out APOE\_Single\_Hits*

We note, that the ‘*v30\_INT\_Rsid\_FIS.fam’* file is the modified fam file with six columns (first five columns - FID, IID,PID, MID, SEX) and the sixth column contains the residuals obtained by regressing fluid intelligence score on ‘Gen\_Sex’, ‘Age’, ‘CollegeEdu’, ‘ALevelEdu’, and the top 10 PCs. Next, we carry out pairwise interaction using the following command:

*plink --bfile APOE\_Single\_Hits --epistasis set-by-set --make-set Set\_file.txt --out APOE\_Single\_Hits\_FIS\_v30*

**Bivariate GWAS:**

The below given R code (example- Fluid intelligence and HDL) was used to carry out bivariate GWAS:

*#Read all covariates*

*FIS=read.table("Out\_FIS.SingleWald.assoc",header=TRUE)*

*HDL=read.table("Out\_HDL.SingleWald.assoc",header=TRUE)*

*#Calcuate Z statistic*

*FIS$FIS\_Z=FIS$Beta/FIS$SE*

*HDL$HDL\_Z=HDL$Beta/HDL$SE*

*#Get P values*

*FIS$FIS\_P=FIS$Pvalue*

*HDL$HDL\_P=HDL$Pvalue*

*#Merge and make bivariate datasets; omit NA*

*library(dplyr)*

*#FISxHDL*

*FIS\_HDL=left\_join(FIS[,c("Test","FIS\_Z","FIS\_P")],HDL[,c("Test","HDL\_Z","HDL\_P")],by="Test")*

*#head(FIS\_HDL)*

*which(is.na(FIS\_HDL$Test)==TRUE)*

*FIS\_HDL=na.omit(FIS\_HDL)*

*#Bivariate Z matrix prep*

*#HDL*

*Z\_FIS\_HDL=data.matrix(FIS\_HDL[,c("FIS\_Z","HDL\_Z")])*

*rownames(Z\_FIS\_HDL)=FIS\_HDL$Test*

*#Bivariate P matrix prep*

*#HDL*

*P\_FIS\_HDL=data.matrix(FIS\_HDL[,c("FIS\_P","HDL\_P")])*

*rownames(P\_FIS\_HDL)=FIS\_HDL$Test*

*#Calculation of Correlation coefficient*

*source("../../metaUSAT-master/metaUSAT\_v1.17.R")*

*R\_FIS\_HDL=cor.pearson(Z\_FIS\_HDL,P\_FIS\_HDL,p.threshold=10^-5)*

*#METAMANOVA*

*library(foreach)*

*library(doParallel)*

*registerDoParallel(72)*

*#HDL*

*res\_FIS\_HDL=foreach(i=1:nrow(Z\_FIS\_HDL),.combine=rbind) %dopar% metausat(Z\_FIS\_HDL[i,],R\_FIS\_HDL,metamanova=TRUE)*

*rownames(res\_FIS\_HDL)=rownames(Z\_FIS\_HDL)*

*head(res\_FIS\_HDL,n=5)*

*write.table(res\_FIS\_HDL,"FIS\_HDL\_metamanova\_Assoc.txt",sep="\t",quote=FALSE)*

**Source codes for figures**

**Figure 2:**

*data=read.csv(“Heat\_V2.csv")*

*head(data)*

*library(reshape2)*

*library(data.table)*

*setDT(data)*

*data2=dcast(data,Mapped.Gene~GWAS\_Trait,value.var="minus\_log10p")*

*head(data2)*

*data3=as.matrix(data2[,-c(1,2)])*

*rownames(data3)=data2$Mapped.Gene*

*head(data3)*

*library(RColorBrewer)*

*#coul <- colorRampPalette(brewer.pal(8, "YlOrRd"))(25)*

*colSide <- brewer.pal(9, "Set1")*

*heatmap(t(data3),margins = c(7,7),main="Effects of identified genetic loci on other traits")*

*head(data2)*

*library(ggplot2)*

*plt <- ggplot(data,aes(x=Mapped.Gene,y=GWAS\_Trait,fill=minus\_log10p))+theme\_grey()+scale\_fill\_distiller(palette = "RdPu",direction=1,na.value="grey92")*

*plt <- plt + geom\_tile()+theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))*

*plt <- plt + xlab("Mapped Gene") + ylab ("GWAS Trait")+*

*ggtitle("Effects of Identified SNPs on related Traits")+*

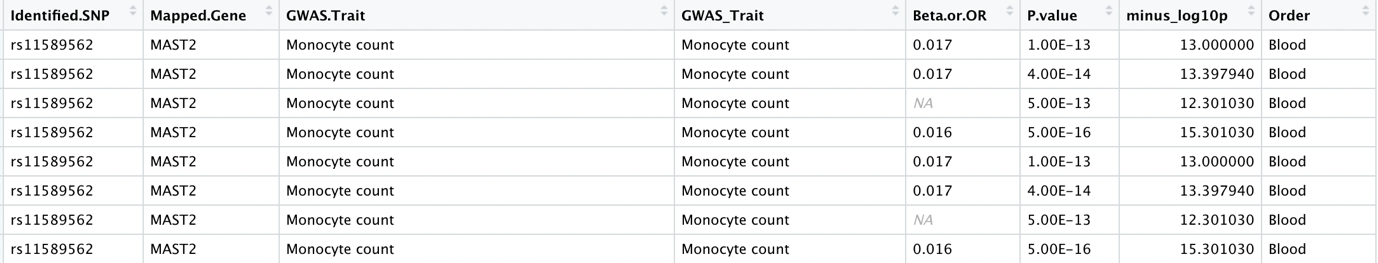
*labs(fill =expression("-log"[10]\*" p"))+ theme(axis.ticks = element\_blank())+*

*theme(panel.grid.major = element\_blank(), panel.grid.minor = element\_blank())+*

*theme(axis.text.x = element\_text(face = "italic"))*

*plt*

Here’s a snippet of the “Heat\_V2.csv” file.

**

**Figure 3B:**

*#install.packages("RCircos")*

*library(RCircos)*

*data("UCSC.HG38.Human.CytoBandIdeogram")*

*#chr.exclude=c("chrX","chrY")*

*chr.exclude <-c("chr3","chr4","chr5","chr7","chr8","chr9","chr10","chr11",*

*"chr13","chr14","chr15","chr16","chr17","chr18","chr20","chr21","chr22","chrX","chrY");*

*cyto.info <- UCSC.HG38.Human.CytoBandIdeogram;*

*RCircos.Set.Plot.Area()*

*#plot.new()*

*RCircos.Set.Core.Components(cyto.info, chr.exclude, tracks.inside=2, tracks.outside=1);*

*RCircos.Chromosome.Ideogram.Plot();*

*params <- RCircos.Get.Plot.Parameters()*

*params$plot.radius=5#default 0.4*

*#RCircos.Reset.Plot.Parameters(params)*

*Link\_data=read.csv(file = Link\_EPi.csv")*

*Gene\_Label\_data=read.csv(file = Gene\_names\_circos.csv")*

*Link\_data1=Link\_data[-6,]*

*Link\_data2=Link\_data[6,]#APOC1*

*Link\_data2$chromStart.1=Link\_data2$chromStart.1+100000*

*Link\_data2$chromEnd.1=Link\_data2$chromStart.1+100000*

*APOE\_lab=read.csv(file = APOE\_loc.csv",header=TRUE)*

*RCircos.Link.Plot(Link\_data1, track.num=1, by.chromosome=FALSE,lineWidth=rep(5,nrow(Link\_data1)),*

*is.sorted=FALSE);*

*RCircos.Link.Plot(Link\_data2, track.num=1, by.chromosome=FALSE,lineWidth=rep(5,nrow(Link\_data2)),*

*is.sorted=FALSE);*

*RCircos.Validate.Genomic.Data(Link\_data, "link")*

*track.num <- 2;*

*new\_params <- RCircos.Get.Plot.Parameters()*

*new\_params$text.size[1]=0.781 #default 0.4*

*RCircos.Reset.Plot.Parameters(new\_params)*

*RCircos.Gene.Name.Plot(Gene\_Label\_data, "Gene", 1, "in");*

*RCircos.Gene.Name.Plot(APOE\_lab, "Gene", 1, "out");*

*legend("bottomright",c("Episodic memory","Simple processing speed","Visual attention"),*

*col=c("#D192B2","goldenrod1","darkturquoise"),lty=1,lwd=3,cex=1.2, bty="n")*

Given below is a snippet of “*Link\_EPi.csv”* obtained from Tables 2 and 3 (main text)

Table

Description automatically generated

The columns of ‘*Gene\_names\_circos.csv*’ are



**Figure 5:**

*library("scatterplot3d")*

*lof=read.csv(“lof\_3d.csv")*

*head(lof)*

*lof$Allele\_Frequency\_Bins=factor(lof$Allele.Frequency.Bins,levels=c("> 5%","1%-5%","0.5%-1%","0.1%-0.5%"))*

*head(lof)*

*library(ggplot2)*

*ggplot(lof, aes(x=LofTool, y=100\*X...of.Variance.Explained,size=12\*Effect.Allele.Freq)) +*

*geom\_point(aes(col=Allele\_Frequency\_Bins))+guides(size="none")+*

*labs(x="LofTool Scores", y="% of variance explained", col="Allele Frequency")+*

*theme(axis.title.x = element\_text(size=14),axis.title.y = element\_text(size=14))+*

*scale\_y\_continuous(limits=c(0,2.5,0.05))*

Given below is a snippet of *‘lof\_3d.csv’* that was obtained from Table 1 (main text).

*Table

Description automatically generated*